日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日
Date of Application: ２００１年　１月２２日

出願番号
Application Number: 特願２００１－０１３５６５

出願人
Applicant(s): 株式会社東芝

CERTIFIED COPY OF
PRIORITY DOCUMENT

２００１年　９月１３日

特許庁長官
Commissioner,
Japan Patent Office

及川耕造

出証番号　出証特２００１－３０８４６３

【書類名】　特許願
【整理番号】　A000007749
【提出日】　平成13年 1月22日
【あて先】　特許庁長官　殿
【国際特許分類】　G06F 15/00
【発明の名称】　べき乗剰余計算装置、べき乗剰余計算方法及び記録媒体
【請求項の数】　16
【発明者】
　【住所又は居所】　神奈川県川崎市幸区小向東芝町１番地　株式会社東芝研究開発センター内
　【氏名】　新保　淳
【発明者】
　【住所又は居所】　神奈川県川崎市幸区小向東芝町１番地　株式会社東芝研究開発センター内
　【氏名】　池田　華恵
【特許出願人】
　【識別番号】　000003078
　【氏名又は名称】　株式会社　東芝
【代理人】
　【識別番号】　100058479
　【弁理士】
　【氏名又は名称】　鈴江　武彦
　【電話番号】　03-3502-3181
【選任した代理人】
　【識別番号】　100084618
　【弁理士】
　【氏名又は名称】　村松　貞男
【選任した代理人】
　【識別番号】　100068814

【弁理士】

【氏名又は名称】　坪井　淳

【選任した代理人】

【識別番号】　100092196

【弁理士】

【氏名又は名称】　橋本　良郎

【選任した代理人】

【識別番号】　100091351

【弁理士】

【氏名又は名称】　河野　哲

【選任した代理人】

【識別番号】　100088683

【弁理士】

【氏名又は名称】　中村　誠

【選任した代理人】

【識別番号】　100070437

【弁理士】

【氏名又は名称】　河井　将次

【手数料の表示】

【予納台帳番号】　011567

【納付金額】　21,000円

【提出物件の目録】

【物件名】　明細書　1

【物件名】　図面　1

【物件名】　要約書　1

【プルーフの要否】　要

【書類名】　明細書

【発明の名称】　べき乗剰余計算装置、べき乗剰余計算方法及び記録媒体

【特許請求の範囲】

【請求項１】

対象データＣ及び独立パラメータｐ、ｑ、ｄについて、複数の整数の組からなる第１の基底及び第２の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第１の基底の全整数の積Ａ＞ｐ，ｑ、且つ、第２の基底の全整数の積Ｂ＞ｐ，ｑ、且つ、Ａ＊Ｂ＞Ｃとする）、計算結果ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求めるためのべき乗剰余計算装置であって、

前記データＣのｐによる剰余値Ｃｐの剰余系表現及び前記パラメータｄの（ｐ－１）による剰余値ｄｐに基づき、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求めるための第１の処理手段と、

前記データＣのｑによる剰余値Ｃｑの剰余系表現及び前記パラメータｄの（ｑ－１）による剰余値ｄｑに基づき、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれにｑが加えられた値の剰余系表現を求めるための第２の処理手段と、

前記第１及び第２の処理手段により求められた両剰余系表現に基づき、法ｐ＊ｑの元でＣ＾ｄと合同である整数ｍ’の剰余系表現を求めるための第３の処理手段と、

前記第３の処理手段により求められた前記剰余系表現を２進数表現に変換して得られた前記整数ｍ’の値に基づき、前記計算結果ｍを求めるための第４の処理手段とを備えたことを特徴とするべき乗剰余計算装置。

【請求項２】

前記第１の処理手段は、前記剰余値Ｃｐ＝Ｃ　ｍｏｄ　ｐの剰余系表現とＢ＾２　ｍｏｄ　ｐの剰余系表現とのＲＮＳモンゴメリ乗算を行い、これにより得られた剰余系表現について指数部を前記剰余値ｄｐ＝ｄ　ｍｏｄ　（ｐ－１）とするＲＮＳモンゴメリべき乗を行うことによって、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求め、

前記第２の処理手段は、前記剰余値Ｃｑ＝Ｃ　ｍｏｄ　ｑの剰余系表現とＢ＾２　ｍｏｄ　ｑの剰余系表現とのＲＮＳモンゴメリ乗算を行い、これにより得ら

出証特２００１－３０８４６３４

れた剰余系表現について指数部を前記剰余値ｄｑ＝ｄ　ｍｏｄ　（ｑ－１）とするＲＮＳモンゴメリべき乗を行うことによって、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれにｑが加えられた値の剰余系表現を求めることを特徴とする請求項１に記載のべき乗剰余計算装置。

【請求項３】

前記パラメータｐ、ｑ及びｄをもとに、前記剰余値ｄｐ＝ｄ　ｍｏｄ　（ｐ－１）及び前記剰余値ｄｑ＝ｄ　ｍｏｄ　（ｑ－１）を求めるための手段を更に備えたことを特徴とする請求項２に記載のべき乗剰余計算装置。

【請求項４】

前記第３の処理手段は、前記第１の処理手段により得られた前記剰余系表現と前記パラメータｑの法ｐにおける逆元ｑｉｎｖ＝ｑ＾（－１）　ｍｏｄ　ｐの剰余系表現とのＲＮＳモンゴメリ乗算を行い、これによって得られた剰余系表現と前記パラメータｑの剰余系表現とのＲＮＳ乗算を行うとともに、前記第２の処理手段により得られた前記剰余系表現と前記パラメータｐの法ｑにおける逆元ｐｉｎｖ＝ｐ＾（－１）　ｍｏｄ　ｑの剰余系表現とのＲＮＳモンゴメリ乗算を行い、これによって得られた剰余系表現と前記パラメータｐの剰余系表現とのＲＮＳ乗算を行い、これらにより得られた両ＲＮＳ乗算の結果のＲＮＳ加算を行うことによって、前記法ｐ＊ｑにおいてＣ＾ｄと合同である整数ｍ’の剰余系表現を求めることを特徴とする請求項１に記載のべき乗剰余計算装置。

【請求項５】

前記パラメータｐ及び前記パラメータｑ並びに前記逆元ｐｉｎｖ及び前記逆元ｑｉｎｖをそれぞれ２進数表現から剰余系表現に変換するための手段を更に備えたことを特徴とする請求項４に記載のべき乗剰余計算装置。

【請求項６】

前記パラメータｐ及びｑをもとに、前記逆元ｐｉｎｖ＝ｐ＾（－１）　ｍｏｄ　ｑ及び前記パラメータｑの法ｐにおける逆元ｑｉｎｖ＝ｑ＾（－１）　ｍｏｄ　ｐを求めるための手段を更に備えたことを特徴とする請求項５に記載のべき乗剰余計算装置。

【請求項７】

前記データＣ並びに前記パラメータｐ及びｑをもとに、前記剰余値Ｃｐ＝Ｃ　ｍｏｄ　ｐ及び前記剰余値Ｃｑ＝Ｃ　ｍｏｄ　ｑを求めるための手段を更に備えたことを特徴とする請求項１に記載のべき乗剰余計算装置。

【請求項８】

前記パラメータｐ、ｑ、ｄのみに依存する剰余系表現のデータを予め記憶しておくための記憶手段を更に備えたことを特徴とする請求項１、２または４に記載のべき乗剰余計算装置。

【請求項９】

前記パラメータを識別する識別情報ｉと、該識別情報ｉに対応するパラメータｐｉ、ｑｉ、ｄｉのみに依存する剰余系表現のデータとを対応付けて予め記憶しておくための記憶手段を更に備えたことを特徴とする請求項１、２または４に記載のべき乗剰余計算装置。

【請求項１０】

前記第１の処理手段及び第２の処理手段はそれらの処理の少なくとも一部を同時に実行することを特徴とする請求項１に記載のべき乗剰余計算装置。

【請求項１１】

前記第１の処理手段及び第２の処理手段は前記基底の要素ごとに行う演算につき該要素の数に相当する演算の全部または一部を同時に実行することを特徴とする請求項１に記載のべき乗剰余計算装置。

【請求項１２】

前記第４の処理手段は、

前記第３の処理手段により求められた前記整数ｍ’の剰余系表現を２進数表現に変換するための手段と、

この手段により得られたｐ＊ｑ未満の前記整数ｍ’の値又はｐ＊ｑ以上の前記整数ｍ’から所定回数ｐ＊ｑを減じることによって得たｐ＊ｑ未満の値を、ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑとするための手段とを含むことを特徴とする請求項１に記載のべき乗剰余計算装置。

【請求項１３】

前記第１の基底の要素数と前記第２の基底の要素数とを同一にしたことを特徴

とする請求項１に記載のべき乗剰余計算装置。

【請求項１４】

対象データＣ及び独立パラメータｐ、ｑ、ｄについて、複数の整数の組からなる第１の基底及び第２の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第１の基底の全整数の積Ａ＞ｐ，ｑ、且つ、第２の基底の全整数の積Ｂ＞ｐ，ｑ、且つ、Ａ＊Ｂ＞Ｃとする）、ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求めるためのべき乗剰余計算方法であって、

前記データＣのｐによる剰余値Ｃｐの剰余系表現及び前記パラメータｄの（ｐ－１）による剰余値ｄｐに基づき、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求めるとともに、前記データＣのｑによる剰余値Ｃｑの剰余系表現及び前記パラメータｄの（ｑ－１）による剰余値ｄｑに基づき、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれにｑが加えられた値の剰余系表現を求め、

求められた両剰余系表現に基づき、法ｐ＊ｑの元でＣ＾ｄと合同である整数ｍ’の剰余系表現を求め、

求められた前記剰余系表現を２進数表現に変換して得られた前記整数ｍ’の値に基づき、前記計算結果ｍを求めることを特徴とするべき乗剰余計算方法。

【請求項１５】

対象データＣ及び独立パラメータｐ、ｑ、ｄについて、複数の整数の組からなる第１の基底及び第２の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第１の基底の全整数の積Ａ＞ｐ，ｑ、且つ、第２の基底の全整数の積Ｂ＞ｐ，ｑ、且つ、Ａ＊Ｂ＞Ｃとする）、ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求めるためのべき乗剰余計算装置としてコンピュータを機能させるためのプログラムを記録したコンピュータ読取り可能な記録媒体であって、

前記データＣのｐによる剰余値Ｃｐの剰余系表現及び前記パラメータｄの（ｐ－１）による剰余値ｄｐに基づき、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求めるための第１の処理機能と、

前記データＣのｑによる剰余値Ｃｑの剰余系表現及び前記パラメータｄの（ｑ－１）による剰余値ｄｑに基づき、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれに

ｑが加えられた値の剰余系表現を求めるための第２の処理機能と、

前記第１及び第２の処理機能により求められた両剰余系表現に基づき、法ｐ＊ｑの元でＣ＾ｄと合同である整数ｍ’の剰余系表現を求めるための第３の処理機能と、

前記第３の処理機能により求められた前記剰余系表現を２進数表現に変換して得られた前記整数ｍ’の値に基づき、前記計算結果ｍを求めるための第４の処理機能とをコンピュータに実現させるためのプログラムを記録したコンピュータ読取り可能な記録媒体。

【請求項１６】

対象データＣ及び独立パラメータｐ、ｑ、ｄについて、複数の整数の組からなる第１の基底及び第２の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第１の基底の全整数の積Ａ＞ｐ，ｑ、且つ、第２の基底の全整数の積Ｂ＞ｐ，ｑ、且つ、Ａ＊Ｂ＞Ｃとする）、ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求めるためのべき乗剰余計算装置としてコンピュータを機能させるためのプログラム、

前記データＣのｐによる剰余値Ｃｐの剰余系表現及び前記パラメータｄの（ｐ－１）による剰余値ｄｐに基づき、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求めるための第１の処理機能と、

前記データＣのｑによる剰余値Ｃｑの剰余系表現及び前記パラメータｄの（ｑ－１）による剰余値ｄｑに基づき、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれにｑが加えられた値の剰余系表現を求めるための第２の処理機能と、

前記第１及び第２の処理機能により求められた両剰余系表現に基づき、法ｐ＊ｑの元でＣ＾ｄと合同である整数ｍ’の剰余系表現を求めるための第３の処理機能と、

前記第３の処理機能により求められた前記剰余系表現を２進数表現に変換して得られた前記整数ｍ’の値に基づき、前記計算結果ｍを求めるための第４の処理機能とをコンピュータに実現させるためのプログラム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、対象データＣ及び独立パラメータｐ、ｑ、ｄについてｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求めるためのべき乗剰余計算装置及びべき乗剰余計算方法に関する。

【０００２】

【従来の技術】

整数演算（加減乗算）の並列処理が可能となる剰余系（Ｒｅｓｉｄｕｅ　Ｎｕｍｂｅｒ　Ｓｙｓｔｅｍ：ＲＮＳ）表現をベースに、公開鍵暗号のアルゴリズム（べき乗剰余演算）を実現するための基本要素となる剰余乗算を、モンゴメリ乗算と融合させて実現するアルゴリズムおよびそのハードウェア構成が提案されている。これをＲＮＳモンゴメリ乗算と呼ぶことにする。

【０００３】

ここで、ＲＮＳ表現（剰余系表現）について説明する。ＲＳＡ暗号などの公開鍵暗号の多くでは多倍長整数を利用して変換を行うが、多倍長整数の表現に通常利用されるのは基数を２とした記数法（Ｒａｄｉｘ　ｒｅｐｒｅｓｅｎｔａｔｉｏｎ）、いわゆる２進数表現である。これとは別の方法に、複数の法a_1，a_2，…，a_nを用意し、整数ｘをそれらの法による剰余値x_1，x_2，…，x_nの組で表現する方法がある。すなわち、以下の式による。

$x_1 = x \mod a_1$，$x_2 = x \mod a_2$，…，$x_n = x \mod a_n$

この表現法はＲＮＳ表現と呼ばれる。

【０００４】

以下では、ＲＮＳ表現で用いる法の集合を、基底（ｂａｓｅ）と呼ぶものとする。また、基底の要素数ｎを基底サイズと呼ぶものとする。基底サイズｎの基底ａは、次のように表される。

$a = \{a_1, a_2, \cdots, a_n\}$

ＲＮＳ表現では、各基底の要素は、通常、互いに素な正整数を用い、中国剰余定理により“基底の要素の積”未満の正整数はＲＮＳ表現により一意に表現できることが保証される。すなわち、

基底ａ＝｛a_1，a_2，…，a_n｝

基底ａの要素の積Ａ＝a_1＊a_2＊…＊a_n

としたときに、基底ａを用いたＲＮＳ表現によりＡ未満の正整数を表現できる。

【０００５】

以下では、基底ａを用いてＲＮＳ表現された整数ｘを、＜ｘ＞__ａで表すものとする（基底を省略して＜ｘ＞で表すこともある）。すなわち、

＜ｘ＞__ａ＝（x_1，x_2，…，x_n）

＝（x mod a_1，x mod a_2，…，x mod a_n）

である。

【０００６】

なお、以下の演算で、２種類の基底を用いる場合に、基底ａ＝｛a_1，a_2，…，a_{n1}｝と基底ｂ＝｛b_1，b_2，…，b_{n2}｝について、ａＵｂは、｛a_1，a_2，…，a_{n1}｝と｛b_1，b_2，…，b_{n2}｝の結合を表し、＜ｘ＞__ａＵｂは、基底ａＵｂによるｘのＲＮＳ表現を表す（すなわち、＜ｘ＞__ａＵｂは、＜ｘ＞__ａ＝（x mod a_1，x mod a_2，…，x mod a_{n1}）と＜ｘ＞__ｂ＝（x mod b_1，x mod b_2，…，x mod b_{n2}）の結合を表す）。また、この場合に、２種類の基底ｎ１＝ｎ２＝ｎとして説明する。

【０００７】

ＲＮＳ表現の利点は、基底の全要素の積Ａを法とした加算、減算、乗算が簡単に計算できることである。すなわち、以下のように各要素をそれぞれの法によって独立に加算、減算、乗算した結果が、所望の結果となる。

<x>_a+<y>_a (mod A)=(x_1+y_1 (mod a_1), x_2+y_2 (mod a_2), …, x_n+y_n (mod a_n))

<x>_a−<y>_a (mod A)=(x_1-y_1 (mod a_1), x_2-y_2 (mod a_2), …, x_n-y_n (mod a_n))

<x>_a ＊<y>_a (mod A)=(x_1*y_1 (mod a_1), x_2*y_2 (mod a_2), …, x_n*y_n (mod a_n))

なお、上記の演算を、それぞれ、ＲＮＳ加算、ＲＮＳ減算、ＲＮＳ乗算と呼ぶものとする。

【０００８】

従って、ｎ個の演算はすべて並列に処理可能であるため、ｎ個の演算ユニットを用意すればすべてを並列に処理して高速な処理が行えるし、用意する演算ユニットがｎ個に満たなくても１個からｎ個まで増やす程それに比例して演算速度を向上できる。

【０００９】

次に、ＲＮＳモンゴメリ乗算およびＲＮＳモンゴメリべき乗について説明する。

【００１０】

ＲＮＳモンゴメリ乗算は、法ｐでの剰余付き乗算（＜ｘ＞＊＜ｙ＞　（ｍｏｄ　Ｎ））に関し、モンゴメリ乗算と呼ばれる手法をＲＮＳ表現での演算に応用する手法であり、概ね以下の手順で計算される。

【００１１】

［ＲＮＳモンゴメリ乗算：ＭＭ（　＜ｘ＞＿ａＵｂ，　＜ｙ＞＿ａＵｂ，　Ｎ，　ａＵｂ　）］

入力：＜ｘ＞＿ａＵｂ，　＜ｙ＞＿ａＵｂ，　Ｎ

　　ただし、ｘ，ｙ　＜　２Ｎ

基底：ａ，ｂ

　　ただし、ｘ，ｙ，Ｎ　＜　Ａ，Ｂ

出力：＜ｗ＞＿ａＵｂ

　　ただし、ｗ＝ｘ＊ｙ＊Ｂ＾(-1)　ｍｏｄ　Ｎ

＜処理内容＞

Step-M-０：　＜（－Ｎ＾(-1)）＞＿ｂを計算する。

Step-M-１：　＜ｓ＞＿ａ＝＜ｘ＞＿ａ　＊　＜ｙ＞＿ａを計算する。

Step-M-２：　＜ｓ＞＿ｂ＝＜ｘ＞＿ｂ　＊　＜ｙ＞＿ｂを計算する。

Step-M-３：　＜ｔ＞＿ｂ＝＜ｓ＞＿ｂ　＊　＜（－Ｎ＾(-1)）＞＿ｂを計算する。

Step-M-４：　＜ｔ＞＿ｂを＜ｔ＞＿ａへ基底変換する。

Step-M-５：　＜ｕ＞＿ａ＝＜ｔ＞＿ａ　＊　＜Ｎ＞＿ａを計算する。

Step-M-６：　＜ｖ＞＿ａ＝＜ｓ＞＿ａ　＋　＜ｕ＞＿ａを計算する。

Step-M-７：　＜ｗ＞＿ａ＝＜ｖ＞＿ａ　＊　＜Ｂ＾(-1)＞＿ａを計算する。

Step-M-８：　＜ｗ＞＿ａを＜ｗ＞＿ｂへ基底変換する。

【００１２】

なお、上記の手順のうちStep-M-４やStep-M-８の基底変換は、ある基底によるＲＮＳ表現に対応するある整数（例えば、基底ｂによるＲＮＳ表現＜ｔ＞＿ｂに対応する整数ｔ）の、他の基底によるＲＮＳ表現（例えば、基底ａによるＲＮＳ表現＜ｔ＞＿ａ）を求める処理である。

【００１３】

ＲＮＳモンゴメリ乗算器も並列に処理を行う演算ユニットを増加させることにより、高速処理が可能となる。

【００１４】

また、ＲＮＳモンゴメリ乗算を繰り返し行うことで（ＲＮＳモンゴメリ乗算器を繰り返し利用することで）べき乗計算を行うことによりＲＳＡ暗号の暗号処理を構成する方法が提案されている。このべき乗計算法をＲＮＳモンゴメリべき乗と呼ぶものとする。ＲＮＳモンゴメリべき乗は、概ね以下の手順で計算される。

【００１５】

［ＲＮＳモンゴメリべき乗：ＭＥＸＰ（　＜ｘ＞＿ａＵｂ，　ｄ，　Ｎ，　ａＵｂ　）］

入力：＜ｘ＞＿ａＵｂ，　指数ｄ＝（ｄ＿ｋ，　ｄ＿ｋ－１，　…，　ｄ＿１）＿２、法Ｎ

　　ただし、ｘ　＜　２Ｎ

基底：ａ，ｂ

　　ただし、ｘ，Ｎ　＜　Ａ，Ｂ

出力：＜ｙ＞＿ａＵｂ

　　ただし、ｙ＝ｘ＾ｄ＊Ｂ＾(-(d-1))　ｍｏｄ　Ｎ

＜処理内容＞

Step-E-１：　ｉ＝ｋとする。＜ｙ＞＿ａＵｂ＝＜Ｂ＞＿ａＵｂとする。

Step-E-２：　＜ｙ＞＿ａＵｂ＝ＭＭ（＜ｙ＞＿ａＵｂ，＜ｙ＞＿ａＵｂ，Ｎ

，ａＵｂ）を計算する。

Step-Ｅ-３：　ｄ＿ｉ＝１ならば、＜ｙ＞＿ａＵｂ＝ＭＭ（＜ｙ＞＿ａＵｂ，＜ｘ＞＿ａＵｂ，Ｎ，ａＵｂ）を計算する。ｄ＿ｉ≠１ならば、なにもしない（ｎｏｐ）。

Step-Ｅ-４：　ｉ＝ｉ－１とする。

Step-Ｅ-５：　ｉ＝０ならば、終了する。ｉ≠０ならば、Step-Ｅ-２へ戻る。

【００１６】

なお、上記の手順のうち、Step-Ｅ-２やStep-Ｅ-３でのＭＭ（　）は、先に説明したＲＮＳモンゴメリ乗算を表す。

【００１７】

次に、ＣＲＴべき乗剰余計算について説明する。

【００１８】

ＲＳＡ暗号は、公開鍵（Ｎ、ｅ）、秘密鍵（ｄ、ｐ、ｑ）に対し、Ｃ＝ｍ＾ｅ　ｍｏｄ　Ｎで平文ｍを暗号文Ｃに暗号変換し、ｍ＝Ｃ＾ｄ　ｍｏｄ　Ｎで暗号文Ｃを平文ｍに復号変換する。ここで、公開鍵である法Ｎの秘密素因数ｐ、ｑを利用して復号変換をより効率的に実行する計算法である、中国剰余定理（Ｃｈｉｎｅｓｅ　Ｒｅｍａｉｎｄｅｒ　Ｔｈｅｏｒｅｍ：ＣＲＴ）を利用したべき乗計算法が知られている。このようなべき乗計算法を、ＣＲＴべき乗剰余計算と呼ぶものとする。

【００１９】

＜ＣＲＴべき乗剰余計算手順＞

(Step-C-１)　ｄｐ＝ｄ　ｍｏｄ　（ｐ－１）
　　ｄｑ＝ｄ　ｍｏｄ　（ｑ－１）

(Step-C-２)　Ｃｐ＝Ｃ　ｍｏｄ　ｐ
　　Ｃｑ＝Ｃ　ｍｏｄ　ｑ

(Step-C-３)　ｍｐ＝Ｃｐ＾ｄｐ　ｍｏｄ　ｐ
　　ｍｑ＝Ｃｑ＾ｄｑ　ｍｏｄ　ｑ

(Step-C-４)　ｍ＝ｍｐ＊（ｑ＾(-1)　mod　ｐ）＊ｑ　＋
　　ｍｑ＊（ｐ＾(-1)　mod　ｑ）＊ｐ　(mod　Ｎ)

なお、上記の手順において、ｄｐ、ｄｑ、（ｑ＾(-1)　ｍｏｄ　ｐ）、（ｐ＾(-1)　ｍｏｄ　ｑ）といったパラメータは秘密鍵のみに依存するので、事前に計算し、秘密鍵の一部として記憶しておくことが一般的である。

【００２０】

ＣＲＴべき乗剰余計算の計算量は、その支配的な部分が（Step-C-３）のべき乗剰余計算２回であり、べき乗剰余計算は法のサイズの３乗に比例することに注意すると、２進表現でのべき乗剰余計算とＣＲＴべき乗剰余計算の計算量は約１／４であることが分かる。なお、（Step-C-３）のべき乗計算を２つの計算回路で同時に実行すれば計算時間は約１／８にできる。

【００２１】

しかしながら、ＲＮＳモンゴメリ乗算によるＲＳＡ暗号装置においてＣＲＴべき乗剰余計算の具体的な手順は示されておらず、ＲＳＡ復号変換（秘密変換）を同装置でより短時間に計算することはできなかった。

【００２２】

【発明が解決しようとする課題】

従来、ＲＮＳモンゴメリ乗算によるＣＲＴべき乗剰余計算の具体的な手順が知られていなかったため、ＲＳＡ復号変換（秘密変換）のような大きな整数のべき乗剰余計算をより高速化することが困難であった。

【００２３】

本発明は、上記事情を考慮してなされたもので、べき乗剰余計算をより効率良く実行可能にしたべき乗剰余計算装置及びべき乗剰余計算方法を提供することを目的とする。

【００２４】

【課題を解決するための手段】

本発明は、対象データＣ及び独立パラメータｐ、ｑ、ｄについて、複数の整数の組からなる第１の基底及び第２の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第１の基底の全整数の積Ａ＞ｐ，ｑ、且つ、第２の基底の全整数の積Ｂ＞ｐ，ｑ、且つ、Ａ＊Ｂ＞Ｃとする）、計算結果ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求めるためのべき乗剰余計算装置であって、

前記データＣのｐによる剰余値Ｃｐの剰余系表現及び前記パラメータｄの（ｐ－１）による剰余値ｄｐに基づき、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求めるための第１の処理手段と、前記データＣのｑによる剰余値Ｃｑの剰余系表現及び前記パラメータｄの（ｑ－１）による剰余値ｄｑに基づき、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれにｑが加えられた値の剰余系表現を求めるための第２の処理手段と、前記第１及び第２の処理手段により求められた両剰余系表現に基づき、法ｐ＊ｑの元でＣ＾ｄと合同である整数ｍ’の剰余系表現を求めるための第３の処理手段と、前記第３の処理手段により求められた前記剰余系表現を２進数表現に変換して得られた前記整数ｍ’の値に基づき、前記計算結果ｍを求めるための第４の処理手段とを備えたことを特徴とする。

【００２５】

好ましくは、前記第１の処理手段は、前記剰余値Ｃｐ＝Ｃ　ｍｏｄ　ｐの剰余系表現とＢ＾２　ｍｏｄ　ｐの剰余系表現とのＲＮＳモンゴメリ乗算を行い、これにより得られた剰余系表現について指数部を前記剰余値ｄｐ＝ｄ　ｍｏｄ　（ｐ－１）とするＲＮＳモンゴメリべき乗を行うことによって、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求め、前記第２の処理手段は、前記剰余値Ｃｑ＝Ｃ　ｍｏｄ　ｑの剰余系表現とＢ＾２　ｍｏｄ　ｑの剰余系表現とのＲＮＳモンゴメリ乗算を行い、これにより得られた剰余系表現について指数部を前記剰余値ｄｑ＝ｄ　ｍｏｄ　（ｑ－１）とするＲＮＳモンゴメリべき乗を行うことによって、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれにｑが加えられた値の剰余系表現を求めるようにしてもよい。

【００２６】

好ましくは、前記第３の処理手段は、前記第１の処理手段により得られた前記剰余系表現と前記パラメータｑの法ｐにおける逆元ｑｉｎｖ＝ｑ＾（－１）　ｍｏｄ　ｐの剰余系表現とのＲＮＳモンゴメリ乗算を行い、これによって得られた剰余系表現と前記パラメータｑの剰余系表現とのＲＮＳ乗算を行うとともに、前記第２の処理手段により得られた前記剰余系表現と前記パラメータｐの法ｑにおける逆元ｐｉｎｖ＝ｐ＾（－１）　ｍｏｄ　ｑの剰余系表現とのＲＮＳモンゴメ

リ乗算を行い、これによって得られた剰余系表現と前記パラメータｐの剰余系表現とのＲＮＳ乗算を行い、これらにより得られた両ＲＮＳ乗算の結果のＲＮＳ加算を行うことによって、前記法ｐ＊ｑにおいてＣ＾ｄと合同である整数ｍ’の剰余系表現を求めるようにしてもよい。

【００２７】

好ましくは、前記第１の処理手段及び第２の処理手段はそれらの処理の少なくとも一部を同時に実行するようにしてもよい。

【００２８】

好ましくは、前記第１の処理手段及び第２の処理手段は前記基底の要素ごとに行う演算につき該要素の数に相当する演算の全部または一部を同時に実行するようにしてもよい。

【００２９】

好ましくは、前記第４の処理手段は、前記第３の処理手段により求められた前記整数ｍ’の剰余系表現を２進数表現に変換するための手段と、この手段により得られたｐ＊ｑ未満の前記整数ｍ’の値又はｐ＊ｑ以上の前記整数ｍ’から所定回数ｐ＊ｑを減じることによって得たｐ＊ｑ未満の値を、ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑとするための手段とを含むようにしてもよい。

【００３０】

また、本発明は、対象データＣ及び独立パラメータｐ、ｑ、ｄについて、複数の整数の組からなる第１の基底及び第２の基底による剰余系表現を利用して（ただし、両基底に含まれる全整数は互いに素、且つ、第１の基底の全整数の積Ａ＞ｐ，ｑ、且つ、第２の基底の全整数の積Ｂ＞ｐ，ｑ、且つ、Ａ＊Ｂ＞Ｃとする）、ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求めるためのべき乗剰余計算方法であって、前記データＣのｐによる剰余値Ｃｐの剰余系表現及び前記パラメータｄの（ｐ－１）による剰余値ｄｐに基づき、（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐが加えられた値の剰余系表現を求めるとともに、前記データＣのｑによる剰余値Ｃｑの剰余系表現及び前記パラメータｄの（ｑ－１）による剰余値ｄｑに基づき、（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ又はこれにｑが加えられた値の剰余系表現を求め、求められた両剰余系表現に基づき、法ｐ＊ｑの元でＣ＾ｄと合同である整数

m’の剰余系表現を求め、求められた前記剰余系表現を２進数表現に変換して得られた前記整数m’の値に基づき、前記計算結果mを求めることを特徴とする。

【００３１】

なお、装置に係る本発明は方法に係る発明としても成立し、方法に係る本発明は装置に係る発明としても成立する。

また、装置または方法に係る本発明は、コンピュータに当該発明に相当する手順を実行させるための（あるいはコンピュータを当該発明に相当する手段として機能させるための、あるいはコンピュータに当該発明に相当する機能を実現させるための）プログラムとしても成立し、該プログラムを記録したコンピュータ読取り可能な記録媒体としても成立する。

【００３２】

本発明によれば、中国剰余定理を利用した演算と剰余系を利用した演算とモンゴメリ演算とを融合させることによって、べき乗剰余計算をより効率良く実行することができる。例えば、剰余系表現を利用したべき乗剰余演算装置にてＲＳＡ暗号の復号変換に中国剰余定理を利用した効率の良い実装が可能となり、同復号変換を約１／８の処理量に削減できる。

【００３３】

【発明の実施の形態】

以下、図面を参照しながら発明の実施の形態を説明する。

【００３４】

図１に、本発明の一実施形態に係る計算装置の機能的な構成図を示す。

【００３５】

本計算装置１は、ＲＮＳ表現された整数を演算するＲＮＳ演算器１２、２進数表現での補助的な演算を行う演算部と、外部との入出力を行うための入出力部１１と、全体を制御する制御部１３で構成される。

ＲＮＳ演算器１２には、ＲＮＳ逆元計算部（ＩＮＶ）１２２、ＲＮＳモンゴメリ乗算部（ＭＭ）１２３、ＲＮＳモンゴメリべき乗部（ＭＥＸＰ）１２４、ＲＮＳ乗算部（ＭＵＬ）１２５、ＲＮＳ加算部（ＡＤＤ）１２６、第１の表現変換部（２進数表現→ＲＮＳ表現変換部）１２７、第２の表現変換部（ＲＮＳ表現変換

部→２進数表現）１２８、記憶部１２１が存在する。

２進数表現での補助的な演算部としては、剰余計算部１２９と加減算部１３０が存在する。

以上の各演算部のうち規模面で大半はＲＮＳ演算器１２である。

【００３６】

記憶部１２１は、例えばＲＮＳ表現で利用する基底や事前に計算し装置内に記憶されるパラメータ等を記憶する。

【００３７】

ＲＮＳモンゴメリ乗算部１２３は、先に説明したＲＮＳモンゴメリ乗算（ＭＭ）を行う。

ＲＮＳモンゴメリべき乗部１２４は、先に説明したＲＮＳモンゴメリべき乗（ＭＥＸＰ）を行う。

ＲＮＳ乗算部１２５は、先に説明したＲＮＳ乗算（ＭＵＬ）を行う。

ＲＮＳ加算１２６は、先に説明したＲＮＳ加算（ＡＤＤ）を行う。

第１の表現変換部１２７は、２進数表現からＲＮＳ表現への変換を行う。

第２の表現変換部１２８は、ＲＮＳ表現から２進数表現への変換を行う。

なお、これらについては、例えば、文献「小池，佐野，川村，　“Ｃｏｘ－Ｒｏｗｅｒアーキテクチャによる高速な並列モンゴメリ乗算法”，　ＳＣＩＳ２０００，　Ｂ２２，　２０００」に詳しい。

【００３８】

ＲＮＳ逆元計算部（ＩＮＶ）１２２は、＜ｘ＞＿ａを入力として＜－ｘ＾(-1)＞＿ａを計算する。すなわち、＜ｘ＞＿ａの各基底a_iおよび要素x_iについて、x_i　（ｍｏｄ　a_i）から、$-x_i$ ^(-1)　（ｍｏｄ　a_i）を計算する。具体的には、例えば次の手順で実行する。

＜基底a_iにおける逆元計算＞

（Step０）　基底a_iに対するＣａｒｍｉｃｈａｅｌ関数λ（a_i）を計算し、記憶部（ＲＯＭ）に保存しておく。Ｃａｒｍｉｃｈａｅｌ関数λの具体的な計算式は例えば“岡本龍明、山本博資著、「現代暗号」、産業図書、ｐ．１６”に示されている。λ（a_i）のビットサイズはa_iのビットサイズ以下である。

法a_iと互いに素な全てのｘ（＜a_i）に対して、ｘ＾λ（a_i）＝１　（ｍｏｄ　a_i）になる。ここでは入力ｘとしてＲＳＡ暗号の秘密鍵ｐ，ｑ（素数）もしくはその積Ｎ（２素数の積）を想定しているため、これらは必ず法a_iと互いに素になる。

（Step１）　x_i＾(-1)＝x_i＾（λ（a_i）－１）　（ｍｏｄ　a_i）を演算ユニットでの剰余乗算により計算する。

（Step２）　－x_i＾(-1)＝a_i－x_i＾(-1)により計算する。

以上の計算のうち、（Step１）はＣａｒｍｉｃｈａｅｌ関数λ（a_i）のビットサイズがa_iのビットサイズ以下であるため、演算ユニットのワード数を３２ビットとすると、６４回以下のワードサイズの剰余乗算に相当する。

【００３９】

剰余計算部１２９は、２進数表現の被除数ｘと除数ｙを入力して、ｘ　（ｍｏｄ　ｙ）を計算する。この計算手順は通常の除算で実行可能であり、例えば“Ｋｎｕｔｈ著、中川圭介訳、「準数値算法／算術演算」、サイエンス社、ｐ．９０”に示されている。概ね、ｘ１＊ｘ２と同じ計算量である。

加減算部１３０は、２進数の加減算等を行う。

本計算装置１は、次のＲＮＳ演算を組み合わせてＣＲＴべき乗を実行する。

・ＲＮＳモンゴメリ乗算＜ｚ＞＝ＭＭ（＜ｘ＞_ａＵｂ，＜ｙ＞_ａＵｂ，ｐ，ａＵｂ）

ここで、ｚ＝ｘ＊ｙ＊Ａ＾(-1)　ｍｏｄ　ｐ　　ｏｒ　　ｚ＝（ｘ＊ｙ＊Ａ＾(-1)　ｍｏｄ　ｐ）＋ｐに相当する。

・ＲＮＳモンゴメリべき乗＜ｚ＞＝ＭＥＸＰ（＜ｘ＞_ａＵｂ，ｅ，ｐ，ａＵｂ）

ここで、ｚ＝ｘ＾ｅ＊Ａ＾(-(e-1))　ｍｏｄ　ｐ　　ｏｒ　　ｚ＝（ｘ＾ｅ＊Ａ＾(-(e-1))　ｍｏｄ　ｐ）＋ｐに相当する。

・ＲＮＳ乗算＜ｚ＞＝ＭＵＬ（＜ｘ＞_ａＵｂ，＜ｙ＞_ａＵｂ，ａ）

ここで、ｚ＝ｘ＊ｙ　ｍｏｄ　Ａ（基底ａでのｘとｙの乗算）に相当する。

・ＲＮＳ加算＜ｚ＞＝ＡＤＤ（＜ｘ＞_ａＵｂ，＜ｙ＞_ａＵｂ，ａ）

ここで、ｚ＝ｘ＋ｙ　ｍｏｄ　Ａ（基底ａでのｘとｙの加算）に相当する。

【００４０】

上記ＲＮＳ演算における最後の引数（ａやａＵｂなど）は、ＲＮＳ表現で利用される基底を表す。基底ａの要素の積の値をＡ、基底ｂの要素の積の値をＢとすると、基底ａＵｂの要素の積の値はＡＢになる。ＲＮＳモンゴメリ乗算とＲＮＳモンゴメリべき乗の出力はｚ＜Ａ，Ｂとなる。

【００４１】

上記のようにＲＮＳモンゴメリ乗算およびＲＮＳモンゴメリべき乗においては、モンゴメリ乗算の性質から、モジュラスｐの値だけ結果が大きい場合がある。すなわち、ＭＭ（＜ｘ＞，＜ｙ＞，ｐ，ａＵｂ）＜２ｐである。法ｐを固定した場合、ＲＮＳモンゴメリ乗算とＲＮＳモンゴメリべき乗の出力はいずれも２ｐ未満であるが、これらの出力はそのままＲＮＳモンゴメリ乗算およびＲＮＳモンゴメリべき乗に入力可能である。

【００４２】

本計算装置１内には、次のパラメータは、予め記憶しておくものとする。

事前登録パラメータ：基底ａ、基底ｂ、基底ａの要素の積Ａ、基底ｂの要素の積Ｂ、基底ａおよび基底ｂの全要素の積ＡＢ、Ｂ＾２

なお、基底ａ，ｂとＣＲＴべき乗におけるパラメータサイズの関係として、少なくとも、ｐ，ｑ＜Ａ、かつ、ｐ，ｑ＜Ｂが必要である。この結果、Ｎ＝ｐ＊ｑに対し、少なくともＮ＜ＡＢとなる。

【００４３】

ＣＲＴべき乗を実行するために外部から本計算装置１へ入力するパラメータは、ここでは、以下の通りとする。

外部入力パラメータ：暗号文Ｃ、ｄｐ＝ｄ　ｍｏｄ　（ｐ－１）、ｄｑ＝ｄ　ｍｏｄ　（ｑ－１）、Ｎ（＝ｐ＊ｑ）、ｐ、ｑ、ｐの法ｑにおける逆元ｐｉｎｖ＝ｐ＾（－１）　ｍｏｄ　ｑ、ｑの法ｐにおける逆元ｑｉｎｖ＝ｑ＾(-1)　ｍｏｄ　ｐ

図２に、本計算装置１におけるＣＲＴべき乗の処理手順の一例を示す。また、図３に、本計算装置１の各演算部に関する内部構成例を示す。

【００４４】

（ステップＳ０）：外部入力パラメータＣ、ｄｐ、ｄｑ、Ｎ、ｐ、ｑ、ｐｉｎｖ、ｑｉｎｖを入力する。

【００４５】

以下の手順のうち、ステップＳ１－ｐ～Ｓ９－ｐと、ステップＳ１－ｑ～Ｓ９－ｑとでは、いずれの対応するステップＳｉ－ｐとステップＳｉ－ｑにおいても、Ｎの２つの素因数であるｐとｑに関する同様の演算を実行している。

【００４６】

（ステップＳ１－ｐ）

第１の表現変換部１２７を利用して、２進数表現ｐを、基底ａＵｂによるＲＮＳ表現＜ｐ＞（＝＜ｐ＞＿ａ　Ｕ　＜ｐ＞＿ｂ＝｛ｐ　ｍｏｄ　a_1，ｐ　ｍｏｄ　a_2，…，ｐ　ｍｏｄ　a_n｝　Ｕ　｛ｐ　ｍｏｄ　b_1，ｐ　ｍｏｄ　b_2，…，ｐ　ｍｏｄ　b_n｝）に変換する。

【００４７】

（ステップＳ１－ｑ）

第１の表現変換部１２７を利用して、２進数表現ｑを、基底ａＵｂによるＲＮＳ表現＜ｑ＞（＝＜ｑ＞＿ａ　Ｕ　＜ｑ＞＿ｂ＝｛ｑ　ｍｏｄ　a_1，ｑ　ｍｏｄ　a_2，…，ｑ　ｍｏｄ　a_n｝　Ｕ　｛ｑ　ｍｏｄ　b_1，ｑ　ｍｏｄ　b_2，…，ｑ　ｍｏｄ　b_n｝）に変換する。

【００４８】

（ステップＳ２－ｐ）

ＲＮＳ逆元計算部１２２を利用して、（ステップＳ１－ｐで求められた）＜ｐ＞＿ｂから、＜－ｐ＾(-1)＞＿ｂを計算する。

【００４９】

（ステップＳ２－ｑ）

ＲＮＳ逆元計算部１２２を利用して、（ステップＳ１－ｑで求められた）＜ｑ＞＿ｂから、＜－ｑ＾(-1)＞＿ｂを計算する。

【００５０】

（ステップＳ３－ｐ）

剰余計算部１２９を利用して、ｂｐ＝Ｂ＾２　ｍｏｄ　ｐを計算し、第１の表

現変換部１２７を利用して、ｂｐを２進数表現から基底ａＵｂによるＲＮＳ表現＜ｂｐ＞に変換する。

【００５１】

（ステップＳ３－ｑ）

剰余計算部１２９を利用して、ｂｑ＝Ｂ＾２　ｍｏｄ　ｑを計算し、第１の表現変換部１２７を利用して、ｂｑを２進数表現から基底ａＵｂによるＲＮＳ表現＜ｂｑ＞に変換する。

【００５２】

（ステップＳ４－ｐ）

第１の表現変換部１２７を利用して、ｐｉｎｖを２進数表現から基底ａＵｂによるＲＮＳ表現＜ｐｉｎｖ＞に変換する。

【００５３】

（ステップＳ４－ｑ）

第１の表現変換部１２７を利用して、ｑｉｎｖを２進数表現から基底ａＵｂによるＲＮＳ表現＜ｑｉｎｖ＞に変換する。

【００５４】

（ステップＳ５－ｐ）

剰余計算部１２９を利用して、Ｃｐ＝Ｃ　ｍｏｄ　ｐを計算し、第１の表現変換部１２７を利用して、Ｃｐを２進数表現から基底ａＵｂによるＲＮＳ表現＜Ｃｐ＞に変換する。

【００５５】

（ステップＳ５－ｑ）

剰余計算部１２９を利用して、Ｃｑ＝Ｃ　ｍｏｄ　ｑを計算し、第１の表現変換部１２７を利用して、Ｃｑを２進数表現から基底ａＵｂによるＲＮＳ表現＜Ｃｑ＞に変換する。

【００５６】

（ステップＳ６－ｐ）

ＲＮＳモンゴメリ乗算部１２３を利用して、＜Ｃｐ’＞＝MM（＜Ｃｐ＞，＜ｂｐ＞，　ｐ，　ａＵｂ）を計算する。

＜先に説明したアルゴリズムを利用する場合の処理内容＞

Step-M-１：　＜ｓ＞_ａ＝＜Cp＞_ａ　＊　＜bp＞_ａを計算する。

Step-M-２：　＜ｓ＞_ｂ＝＜Cp＞_ｂ　＊　＜bp＞_ｂを計算する。

Step-M-３：　＜ｔ＞_ｂ＝＜ｓ＞_ｂ　＊　＜（－ｐ^(-1)）＞_ｂを計算する。

Step-M-４：　＜ｔ＞_ｂを＜ｔ＞_ａへ基底変換する。

Step-M-５：　＜ｕ＞_ａ＝＜ｔ＞_ａ　＊　＜ｐ＞_ａを計算する。

Step-M-６：　＜ｖ＞_ａ＝＜ｓ＞_ａ　＋　＜ｕ＞_ａを計算する。

Step-M-７：　＜Cp’＞_ａ＝＜ｖ＞_ａ　＊　＜Ｂ^(-1)＞_ａを計算する。

Step-M-８：　＜Cp’＞_ａを＜Cp’＞_ｂへ基底変換する。

これにより、Ｃｐ’＝Ｃ＊Ｂ　ｍｏｄ　ｐまたはＣｐ’＝（Ｃ＊Ｂ　ｍｏｄ　ｐ）＋ｐのいずれかに対応するＲＮＳ表現＜Ｃｐ’＞が求められる。

【００５７】

（ステップＳ６－ｑ）

ＲＮＳモンゴメリ乗算部１２３を利用して、＜Ｃｑ’＞＝ＭＭ（＜Ｃｑ＞，＜ｂｑ＞，　ｑ，　ａＵｂ）を計算する。なお、先に説明したアルゴリズムを利用する場合の処理内容は、ステップＳ６－ｐの処理内容において、ｐとｑを入れ替えたものである。

これにより、Ｃｑ’＝Ｃ＊Ｂ　ｍｏｄ　ｑまたはＣｑ’＝（Ｃ＊Ｂ　ｍｏｄ　ｑ）＋ｑのいずれかに対応するＲＮＳ表現＜Ｃｐ’＞が求められる。

【００５８】

（ステップＳ７－ｐ）

ＲＮＳモンゴメリべき乗部１２４を利用して、＜ｍｐ’＞＝ＭＥＸＰ（＜Ｃｐ’＞，　ｄｐ，　ｐ，　ａＵｂ）を計算する。

＜先に説明したアルゴリズムを利用する場合の処理内容＞

Step-E-１：　ｉ＝ｋとする。＜ｙ＞_ａＵｂ＝＜Ｂ＞_ａＵｂとする。

Step-E-２：　＜ｙ＞_ａＵｂ＝ＭＭ（＜ｙ＞_ａＵｂ，＜ｙ＞_ａＵｂ，Ｎ，ａＵｂ）を計算する。

Step-E-３：　ｄｐ＿ｉ＝１ならば、＜ｙ＞＿ａＵｂ＝MM（＜ｙ＞＿ａＵｂ，　＜Ｃｐ’＞＿ａＵｂ，ｐ，ａＵｂ）を計算する。ｄｐ＿ｉ≠１ならば、なにもしない（ｎｏｐ）。

ここで、ｄｐ＿ｉは、ｄｐの２進表現（ｄｐ＿ｋ，　ｄｐ＿ｋ－１，…，ｄｐ＿１）における下位からｉビット目の値である。

Step-E-４：　ｉ＝ｉ－１とする。

Step-E-５：　ｉ＝０ならば、終了する。ｉ≠０ならば、Step-E-２へ戻る。

これにより、ｍｐ’＝（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐまたはｍｐ’＝（（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ）＋ｐのいずれかに対応するＲＮＳ表現＜ｍｐ’＞が求められる。

【００５９】

（ステップＳ７－ｑ）

ＲＮＳモンゴメリべき乗部１２４を利用して、＜ｍｑ’＞=ＭＥＸＰ（＜Ｃｑ’＞，　ｄｑ，　ｑ，　ａＵｂ）を計算する。なお、先に説明したアルゴリズムを利用する場合の処理内容は、ステップＳ７－ｐの処理内容において、ｐとｑを入れ替えたものである。

これにより、ｍｑ’＝（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ　またはｍｑ’＝（（Ｃｑ＾ｄｑ）＊Ｂ　ｍｏｄ　ｑ）＋ｑのいずれかに対応するＲＮＳ表現＜ｍｑ’＞が求められる。

【００６０】

（ステップＳ８－ｐ）

ＲＮＳモンゴメリ乗算部１２３を利用して、＜ｔｐ＞=MM（＜ｍｐ’＞，＜ｑ＾（－１）　ｍｏｄ　ｐ＞，　ｐ，　ａＵｂ）を計算する。

＜先に説明したアルゴリズムを利用する場合の処理内容＞

Step-M-１：　＜ｓ＞＿ａ＝＜mp’＞＿ａ　＊　＜qinv＞＿ａを計算する。

Step-M-２：　＜ｓ＞＿ｂ＝＜mp’＞＿ｂ　＊　＜qinv＞＿ｂを計算する。

Step-M-３：　＜ｔ＞＿ｂ＝＜ｓ＞＿ｂ　＊　＜（－ｐ＾(-1)）＞＿ｂを計算する。

Step-M-４：　＜ｔ＞＿ｂを＜ｔ＞＿ａへ基底変換する。

Step-M-５：　＜ｕ＞＿ａ＝＜ｔ＞＿ａ　＊　＜ｐ＞＿ａを計算する。

Step-M-６：　＜ｖ＞＿ａ＝＜ｓ＞＿ａ　＋　＜ｕ＞＿ａを計算する。

Step-M-７：　＜tp＞＿ａ＝＜ｖ＞＿ａ　＊　＜Ｂ＾(-1)＞＿ａを計算する。

Step-M-８：　＜tp＞＿ａを＜tp＞＿ｂへ基底変換する。

これにより、ｔｐ＝（Ｃｐ＾ｄｐ）ｑ＾（－１）　ｍｏｄ　ｐまたはｔｐ＝（（Ｃｐ＾ｄｐ）ｑ＾（－１）　ｍｏｄ　ｐ）＋ｐのいずれかに対応するＲＮＳ表現＜ｔｐ＞が求められる。

【００６１】

（ステップＳ８－ｑ）

ＲＮＳモンゴメリ乗算部１２３を利用して、＜ｔｑ＞＝ＭＭ（＜ｍｑ’＞，＜ｐ＾（－１）　ｍｏｄ　ｑ＞，　ｐ，　ａＵｂ）を計算する。なお、先に説明したアルゴリズムを利用する場合の処理内容は、ステップＳ８－ｐの処理内容において、ｐとｑを入れ替えたものである。

これにより、ｔｑ＝（Ｃｑ＾ｄｑ）ｐ＾（－１）　ｍｏｄ　ｑまたはｔｑ＝（（Ｃｑ＾ｄｑ）ｐ＾（－１）　ｍｏｄ　ｑ）＋ｑのいずれかに対応するＲＮＳ表現＜ｔｑ＞が求められる。

【００６２】

（ステップＳ９－ｐ）

ＲＮＳ乗算部１２５を利用して、＜ｕｐ＞＝ＭＵＬ（＜ｔｐ＞，　＜ｑ＞，ａＵｂ）を計算する。

これにより、ｕｐ＝ｔｐ＊ｑ　ｍｏｄ　ＡＢに対応するＲＮＳ表現＜ｕｐ＞が求められる。

【００６３】

（ステップＳ９－ｑ）

ＲＮＳ乗算部１２５を利用して、＜ｕｑ＞＝ＭＵＬ（＜ｔｑ＞，　＜ｐ＞，ａＵｂ）を計算する。

これにより、ｕｑ＝ｔｑ＊ｐ　ｍｏｄ　ＡＢに対応するＲＮＳ表現＜ｕｑ＞が求められる。

【００６４】

（ステップＳ１０）

ＲＮＳ加算１２６を利用して、＜ｍ’＞＝ＡＤＤ（＜ｕｐ＞，　＜ｕｑ＞，ａＵｂ）を計算する。

これにより、ｍ’＝ｕｐ＋ｕｑ　ｍｏｄ　ＡＢに対応するＲＮＳ表現＜ｍ’＞が求められる。

【００６５】

（ステップＳ１１）

第２の表現変換部１２８を利用して、＜ｍ’＞をＲＮＳ表現（基底ａＵｂ）から２進数表現ｍ’に変換する。

ここで、ｍ’はＮ以上である場合があるので、加減算部１３０は、ｍ’がＮ以上であれば、これをＮ未満の値にするための処理を行う。

（ステップＳ１２）

ｍ’をｍにコピーする（待避させる）。

（ステップＳ１３）

ｍ’＝ｍ’－Ｎを計算する。

（ステップＳ１４）

ｍ’＜０であるか否か判定する。ｍ’＜０でなければ、ステップＳ１２へ戻る。ｍ’＜０であるならば、ループを抜けて、ステップＳ１５へ移る。

（ステップＳ１５）

ｍを出力して、終了する。

なお、ステップＳ１２～ステップＳ１５は、例えば図４のステップＳ２１～ステップＳ２４など、他の手順でもかまわない。

【００６６】

また、Ｎは、外部から入力せずに、加減算部１３０がｐ＊ｑにより求めるようにしても良い。

【００６７】

上記手順において、ステップＳ５－ｐ，Ｓ６－ｐおよびステップＳ５－ｑ，Ｓ６－ｑでは、Ｃｐ’＝Ｃ＊Ｂ　ｍｏｄ　ｐ（＋ｐ）およびＣｑ’＝Ｃ＊Ｂ　ｍｏｄ　ｑ（＋ｑ）を計算しており、先に説明した通常のＣＲＴべき乗での（Step-C

-２）の処理に対応している。

ステップＳ７－ｐおよびステップＳ７－ｑは、通常のＣＲＴべき乗での（Step-C-３）の処理に対応している。

ステップＳ８－ｐ、Ｓ９－ｐ、Ｓ８－ｑ、Ｓ９－ｑ、Ｓ１０は、先に説明した通常のＣＲＴべき乗での（Step-C-４）の処理に対応している。ここでは、（Step-C-４）の処理が次のように変形できることを利用している。

$$m = mp * (q^{(-1)} \bmod p) * q + mq * (p^{(-1)} \bmod q) * p \pmod N$$

$$= \{mp * (q^{(-1)} \bmod p) \bmod p\} * q + \{mq * (p^{(-1)} \bmod q) \bmod q\} * p \pmod N$$

ステップＳ１１の結果であるｍ’は、ＲＮＳモンゴメリ乗算でのｐおよびｑの加算誤差と、加算誤差が無いとしてもＣＲＴべき乗剰余計算ではｍ’＜２Ｎであることから、加算誤差を考慮するとｍ’＜４Ｎとなる。従って、最大でｍ’から３Ｎを引く必要があり、このために必要な補正をステップＳ１２からステップＳ１４で行っている。ｍ’は２進数に変換してあるので正負の判定は容易である。この処理が積に説明した通常のＣＲＴべき乗での（Step-C-４）の処理における法Ｎでの剰余値を求める手順に相当する。

【００６８】

本ＣＲＴべき乗剰余計算での各計算ステップはＲＮＳ演算器で実行可能な演算機能を用いて実行可能である。特にステップＳ７－ｐとステップＳ７－ｑのＲＮＳモンゴメリべき乗が計算処理の大半を占めるが、これらは法ｐ、ｑよりわずかに大きな基底Ａ、Ｂを用いた和集合ＡＵＢを基底として利用できることが重要である。

ＲＮＳモンゴメリ乗算の計算量は、その内部で実行する基底変換の計算量で評価できる。この処理では、１つの基底要素について見た場合にワードサイズでの乗算を基底サイズｎのオーダだけ必要とし、さらにこれを変換元の基底における全ての基底要素について実行する。従って、ＲＮＳモンゴメリ乗算の計算量は、基底サイズｎの２乗のオーダとなる。また、ＲＮＳモンゴメリべき乗の計算量は、ＲＮＳモンゴメリ乗算を指数のビットサイズＬ＿ｅだけ繰り返す処理に相当す

る。よって、ＲＮＳモンゴメリべき乗の計算量はＯ（ｎ＾２＊Ｌ＿ｅ）となる。

具体的に、例えば、１０２４ビットのＲＳＡ暗号を想定する。この場合、秘密鍵ｄ、Ｎ、暗号文Ｃはいずれも１０２４ビットになる。従って、これを、従来のようにＲＮＳ表現でのモンゴメリべき乗で実行する場合、利用する基底ａ’（およびｂ’）は最低でも要素数３３（＝１０２４／３２（ワードサイズ）＋１）となる。一方、本実施形態のようにＣＲＴべき乗で利用する秘密鍵ｄｐ、ｄｑ、ｐ、ｑ、Ｃを法ｐ、ｑで縮小した値Ｃｐ、Ｃｑはいずれも５１２ビットであるため、利用する基底ａ（およびｂ）は最低で要素数１７（＝５１２／３２（ワードサイズ）＋１）となる。最低の基底要素数を利用するのが処理時間の面で最も効率的であるので、この前提でＣＲＴによるべき乗剰余計算とＣＲＴによらないべき乗剰余計算の計算量を比較してみる。ＲＮＳモンゴメリ乗算については、その計算量は、ＣＲＴを用いる場合は、ＣＲＴを用いない場合の１／４となる。べき指数のサイズについては、ＣＲＴを用いる場合にはＣＲＴを用いない場合の１／２になるが、ＣＲＴを用いる場合にはＲＮＳモンゴメリべき乗を２回計算する必要がある。よって、全体としては、本ＣＲＴべき乗剰余計算によれば、従来のＲＮＳモンゴメリべき乗に比べて、約１／４の処理量でＲＳＡ復号変換が実現できる。また、ＲＮＳモンゴメリべき乗を２つの回路で同時に実行すれば、従来のＲＮＳモンゴメリべき乗に比べて、約１／８の処理量でＲＳＡ復号変換が実現できる。

【００６９】

以下では、本実施形態のバリエーションについて説明する。

【００７０】

図２の手順のうち、ステップＳ１－ｐ～Ｓ５－ｐの手順は、ステップＳ２－ｐがステップＳ１－ｐの後になる制約がある以外、どのような順番で行ってもかまわない（剰余計算部１２９や表現変換部１２７を並列処理可能にして、それらの全部または一部を並列処理してもよい）。

【００７１】

また、図２の手順のうち、ステップＳ１－ｐ～Ｓ９－ｐと、ステップＳ１－ｑ～Ｓ９－ｑとでは、いずれも対応するステップＳｉ－ｐとステップＳｉ－ｑでＮ

の２つの素因数であるｐとｑに関する同様の演算を実行している。ステップＳ１－ｐ～Ｓ９－ｐ、Ｓ１－ｑ～Ｓ９－ｑの演算は、ｐパートとｑパートを逐次的に行っても良いし、全てのｐパートを実行後に全てのｑパートを実行するようにしても良い。後者の方が、中間変数のメモリへの退避・復旧が減る分、効率が良くなる可能性がある。

また、ｐパートとｑパートをパイプライン的に処理することも可能である。

また、該当する演算部の全部または一部を並列処理可能にしてｐパートとｑパートとを並列的に実行することも可能である。ｐパートとｑパートとを別々に記述した場合の計算装置１の各演算部に関する内部構成例を図５に示す。

また、例えば、ＲＮＳモンゴメリ乗算部１２３、ＲＮＳモンゴメリべき乗部１２４、ＲＮＳ乗算部１２５、ＲＮＳ加算部１２６の全部、あるいはそれらのうちＲＮＳモンゴメリ乗算部１２３およびＲＮＳモンゴメリべき乗部１２４のみ、あるいはそれらのうちＲＮＳモンゴメリべき乗部１２４のみを、並列処理可能にしてｐパートとｑパートとを並列的に実行することも可能である。

もちろん、各演算部は、ＲＮＳ演算に由来する並列計算を行って高速化を図ることが可能である。この場合に、基底の全ての要素に対する演算を同時に実行するように構成することも可能であるし、基底の一部（例えば、基底サイズの整数分の一に相当する個数）の要素に対する演算を同時に実行するように構成することも可能である。

【００７２】

また、これまで説明した構成例では、ｐｉｎｖ＝ｐ＾（－１）　ｍｏｄ　ｑ、ｑｉｎｖ＝ｑ＾(-1)　ｍｏｄ　ｐを外部から入力する例を示したが、それらをｐ、ｑから計算するようにしてもよい。この場合には、図５に示すように、２進数表現での補助的な演算部として、剰余計算部１２９と加減算部１３０の他に、さらに逆元計算部１３１を設ければよい。

逆元計算部１３１では、２進数表現の整数ｘと法の値ｙを入力して、ｘ＾(-1)（ｍｏｄ　ｙ）を計算する。この計算は拡張ユークリッドの互除法と呼ばれるアルゴリズムで実行されることが多い。例えば、“Ｋｎｕｔｈ著、中川圭介訳、「準数値算法／算術演算」、サイエンス社、ｐ．１６２”に示されている。概ね

、ｙのサイズの剰余乗算１０回分程度の計算量である。

【００７３】

また、これまで説明した構成例では、ｄｐ＝ｄ　ｍｏｄ　（ｐ－１）、ｄｑ＝ｄ　ｍｏｄ　（ｑ－１）を外部から入力する例を示したが、それらをｐ、ｑから計算するようにしてもよい。この計算は、剰余計算部１２９により行うことができる。

【００７４】

ｐｉｎｖ、ｑｉｎｖ、ｄｐ、ｄｑをｐ、ｑから計算するようにした場合の計算装置１の各演算部に関する内部構成例を図６に示す。

【００７５】

また、外部入力パラメータ（暗号文Ｃ、ｄｐ＝ｄ　ｍｏｄ　（ｐ－１）、ｄｑ＝ｄ　ｍｏｄ　（ｑ－１）、Ｎ（＝ｐ＊ｑ）、ｐ、ｑ、ｐｉｎｖ＝ｐ＾（－１）　ｍｏｄ　ｑ、ｑｉｎｖ＝ｑ＾(-1)　ｍｏｄ　ｐ）のうち暗号文Ｃ以外のものは、ＲＳＡの秘密鍵に相当するパラメータであり、それらの全部または一部を本計算装置１内部に記憶しておくことも可能である。この場合には、外部からは、暗号文Ｃと本演算装置１内部の鍵パラメータ群を選択するのに必要な鍵識別情報を入力するようにすればよい。

【００７６】

また、図２のステップＳ１－ｐ～Ｓ４－ｐおよびステップＳ１－ｑ～Ｓ４－ｑで示される計算は、ＲＳＡの秘密鍵（ｐ，ｑ，ｐｉｎｖ，ｑｉｎｖ）のみに依存した計算であるが、ＲＳＡによる暗号文Ｃはセッション毎に異なるのに対し、ＲＳＡ秘密鍵はそれほど変更されないことが多い（ＲＳＡ秘密鍵が不変のシステムもあり得る）。

そこで、ステップＳ１－ｐ～ステップＳ４－ｑまで実行した結果を保存しておき、以後同じＲＳＡ秘密鍵が用いられる限り、ステップＳ１－ｐ～ステップＳ４－ｑまでスキップし、先に保存しておいた結果を利用してステップＳ５－ｐ以降の処理を行うとともに、ＲＳＡ秘密鍵が変更されたときに改めてステップＳ１－ｐ～ステップＳ４－ｑまで実行するようにしてもよい。

また、ＲＳＡ秘密鍵を鍵識別情報で管理する場合には、鍵識別情報に対応付け

て、上記結果を保存しておくようにしてもよい。

【００７７】

また、ＲＳＡ秘密鍵が唯一不変の場合には、外部からはＣのみを入力するものとし、ＲＳＡ秘密鍵のみに依存するデータ（ｐ、ｑ、Ｎ、＜ｐ＞、＜ｑ＞、＜－ｐ＾(-1)＞＿ｂ、＜－ｑ＾(-1)＞＿ｂ、＜ｂｐ＞、＜ｂｑ＞、＜ｐｉｎｖ＞、＜ｑｉｎｖ＞、＜ｂｐ＞、＜ｂｑ＞）は、予め記憶部に記憶しておくようにしてもよい。

また、ＲＳＡ秘密鍵が複数ある場合には、外部からはＣおよび鍵識別情報のみを入力するものとし、ＲＳＡ秘密鍵のみに依存するデータ（ｐ、ｑ、Ｎ、＜ｐ＞、＜ｑ＞、＜－ｐ＾(-1)＞＿ｂ、＜－ｑ＾(-1)＞＿ｂ、＜ｂｐ＞、＜ｂｑ＞、＜ｐｉｎｖ＞、＜ｑｉｎｖ＞、＜ｂｐ＞、＜ｂｑ＞）を鍵識別情報に対応付けて予め記憶部に記憶しておき、外部から入力された鍵識別情報に対応するものを記憶部から読み出して用いるようにしてもよい。

【００７８】

また、２種類の基底を用いる場合に、基底ａ＝｛a_1，a_2，…，a_{n1}｝と基底ｂ＝｛b_1，b_2，…，b_{n2}｝について、ｎ１＝ｎ２＝ｎとして説明したが、ｎ１≠ｎ２とすることも可能である。

【００７９】

なお、以上では、本発明を図８のような復号変換に適用した場合について説明したが、暗号変換（Ｃ＝ｍ＾ｅ　ｍｏｄ　Ｎ）は復号変換（ｍ＝Ｃ＾ｄ　ｍｏｄ　Ｎ）と同様の計算式により表現されるので、もちろん、本発明は暗号変換にも適用可能である（例えば、秘密鍵を持つ装置が、暗号変換するケース）。この場合には、これまでの説明において、暗号文Ｃの代わりに平文ｍを入力とし、指数ｄの代わりに指数ｅを用いればよい。

【００８０】

ここで、本計算装置のハードウェア、ソフトウェア構成について説明する。

本実施の形態では、本計算装置（復号化装置あるいは暗号化装置）をハードウェアにより実現することを想定して説明したが、ソフトウェアとして実現することも可能である。

ハードウェアとして構成する場合、例えば、半導体装置として形成し、演算ボードあるいは演算カードとして、パーソナル・コンピュータなどの計算機に装備する形態がある。計算機がＯＳを用いる場合には、この演算デバイス用のドライバをＯＳに組み込んで用いる形態もある。また、例えば、半導体装置として形成し、ＡＶ機器や家電機器等の装置に備えることも可能である。

ソフトウェアで実現する場合、コンピュータに所定の手段を実行させるための（あるいはコンピュータを所定の手段として機能させるための、あるいはコンピュータに所定の機能を実現させるための）プログラムとして実施することができ、また該プログラムを記録したコンピュータ読取り可能な記録媒体として実施することができる。もちろん、マルチプロセッサやパイプライン処理などの種々の高速化技術を利用することも可能である。

【００８１】

なお、この発明の実施の形態で例示した構成は一例であって、それ以外の構成を排除する趣旨のものではなく、例示した構成の一部を他のもので置き換えたり、例示した構成の一部を省いたり、例示した構成に別の機能あるいは要素を付加したり、それらを組み合わせたりすることなどによって得られる別の構成も可能である。また、例示した構成と論理的に等価な別の構成、例示した構成と論理的に等価な部分を含む別の構成、例示した構成の要部と論理的に等価な別の構成なども可能である。また、例示した構成と同一もしくは類似の目的を達成する別の構成、例示した構成と同一もしくは類似の効果を奏する別の構成なども可能である。

また、この発明の実施の形態で例示した各種構成部分についての各種バリエーションは、適宜組み合わせて実施することが可能である。

また、この発明の実施の形態は、個別装置としての発明、関連を持つ２以上の装置についての発明、システム全体としての発明、個別装置内部の構成部分についての発明、またはそれらに対応する方法の発明等、種々の観点、段階、概念またはカテゴリに係る発明を包含・内在するものである。

従って、この発明の実施の形態に開示した内容からは、例示した構成に限定されることなく発明を抽出することができるものである。

【００８２】

本発明は、上述した実施の形態に限定されるものではなく、その技術的範囲において種々変形して実施することができる。

【００８３】

【発明の効果】

本発明によれば、中国剰余定理を利用した演算と剰余系を利用した演算とモンゴメリ演算とを融合させることによって、べき乗剰余計算をより効率良く実行することができる。

【図面の簡単な説明】

【図１】

本発明の実施の形態に係る計算装置の構成例を示す図

【図２】

同実施形態に係る計算装置の処理手順の一例を示すフローチャート

【図３】

同実施形態に係る計算装置の各演算部に関する内部構成例を示す図

【図４】

同実施形態に係る計算装置の処理手順の他の例を示すフローチャート

【図５】

同実施形態に係る計算装置の各演算部に関する内部構成の他の例を示す図

【図６】

同実施形態に係る計算装置の他の構成例を示す図

【図７】

同実施形態に係る計算装置の各演算部に関する内部構成のさらに他の例を示す図

【図８】

同実施形態に係る計算装置の適用例について説明するための図

【符号の説明】

１…計算装置

１１…入出力部

１２…ＲＮＳ演算器

１３…制御部

１２１…記憶部

１２２…ＲＮＳ逆元計算部

１２３…ＲＮＳモンゴメリ乗算部

１２４…ＲＮＳモンゴメリべき乗部

１２５…ＲＮＳ乗算部

１２６…ＲＮＳ加算部

１２７…第１の表現変換部

１２８…第２の表現変換部

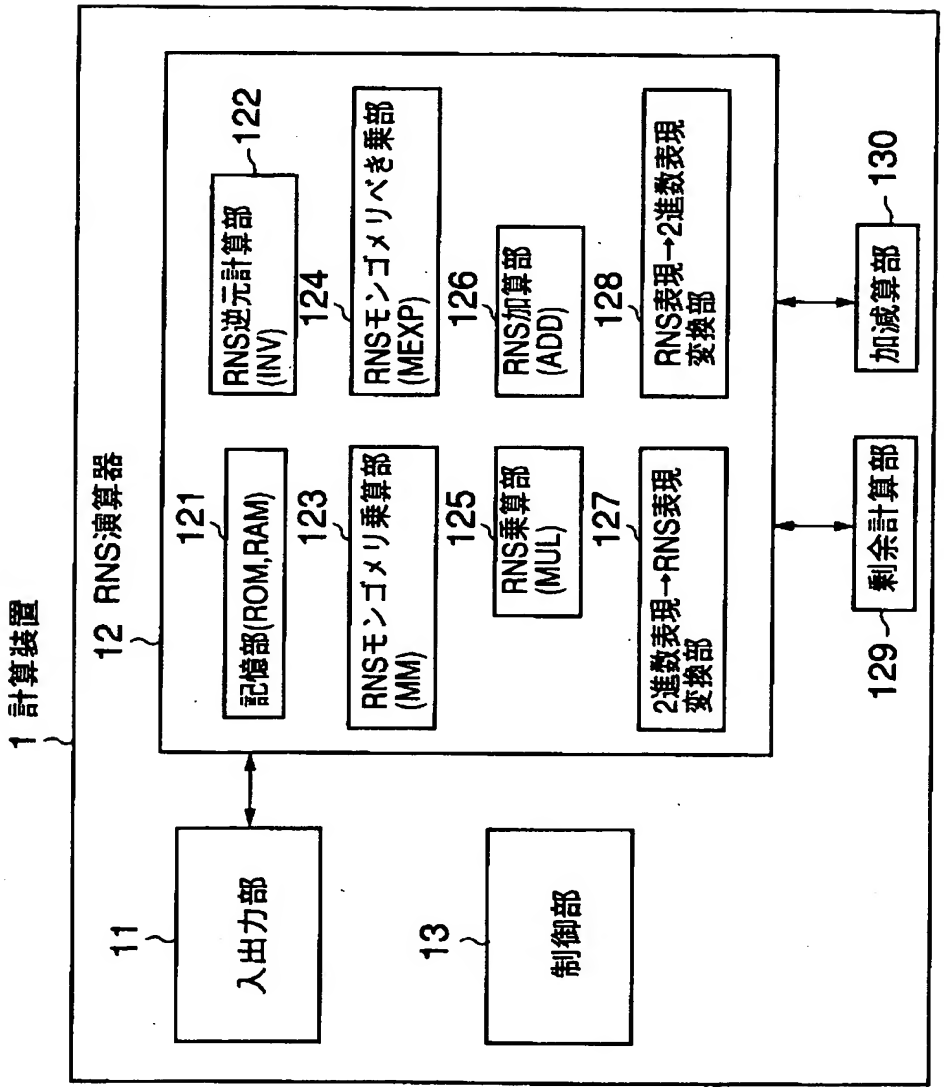
１２９…剰余計算部

１３０…加減算部

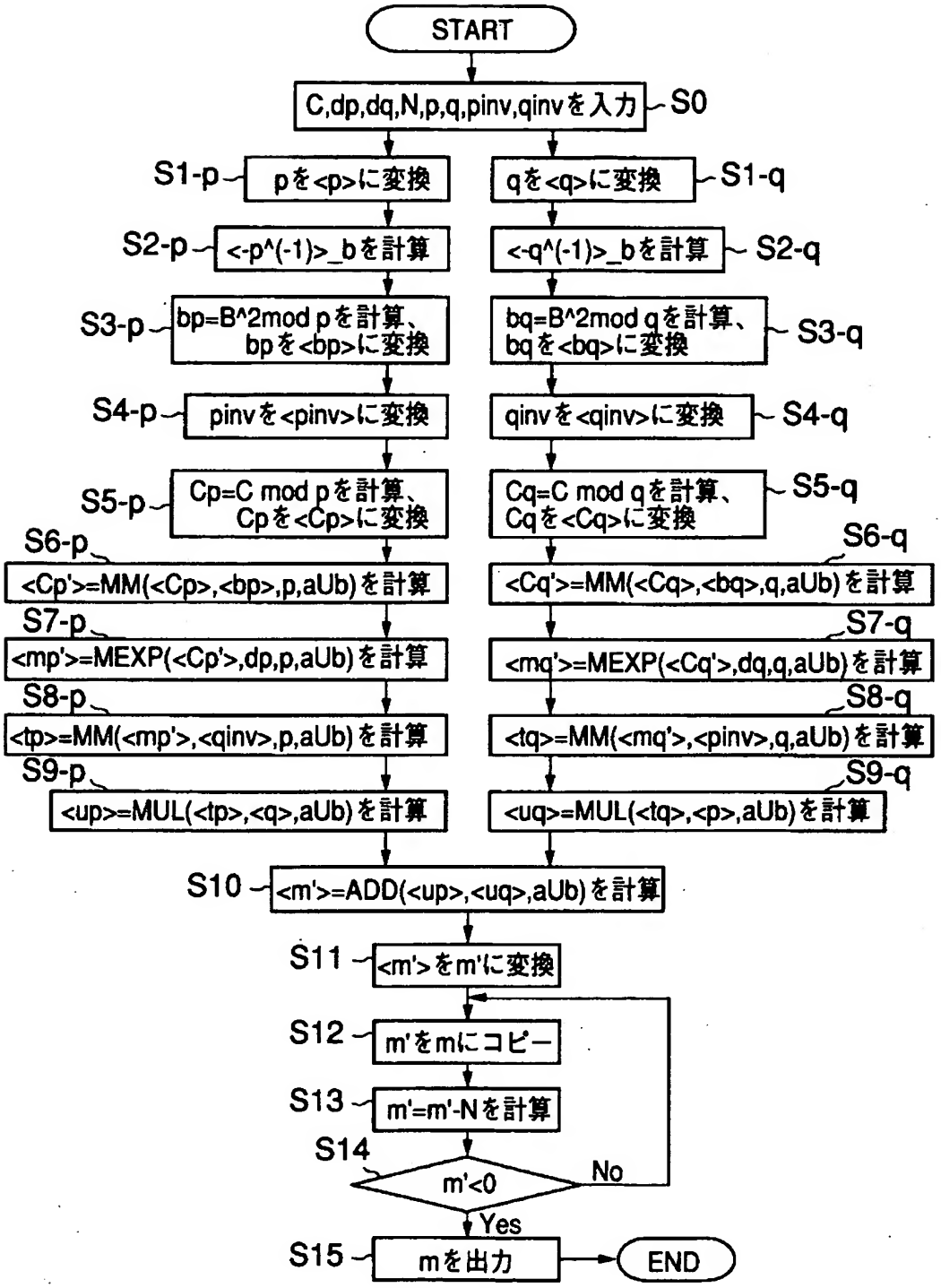
１３１…逆元計算部

【書類名】　　図面

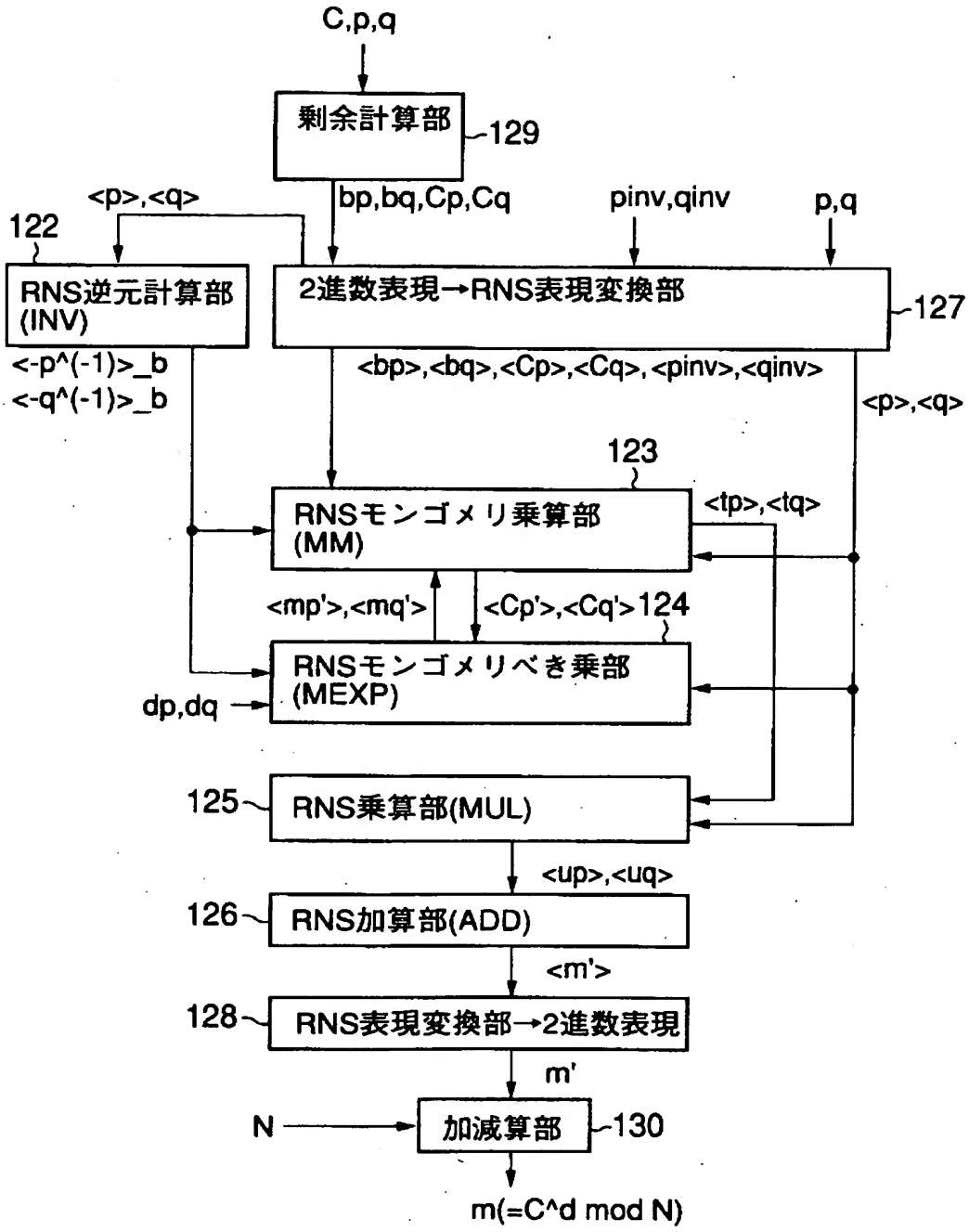
【図１】


1 計算装置
12 RNS演算器
11
入出力部
13
制御部
121
記憶部(ROM,RAM)
122
RNS逆元計算部
(INV)
123
RNSモンゴメリ乗算部
(MM)
124
RNSモンゴメリべき乗部
(MEXP)
125
RNS乗算部
(MUL)
126
RNS加算部
(ADD)
127
2進数表現→RNS表現
変換部
128
RNS表現→2進数表現
変換部
129
剰余計算部
130
加減算部

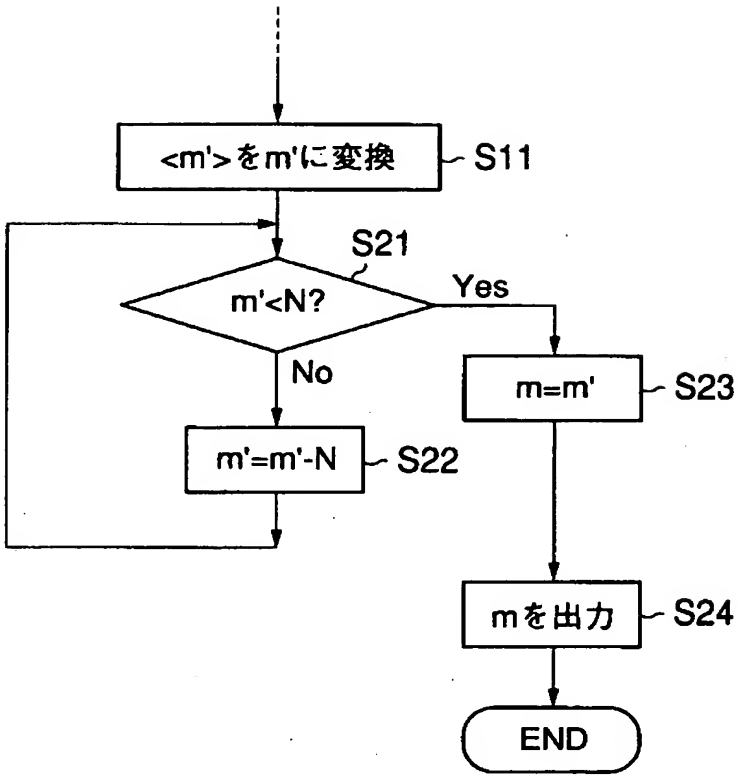
【図２】


START
C,dp,dq,N,p,q,pinv,qinvを入力
S0
S1-p
pを<p>に変換
qを<q>に変換
S1-q
S2-p
<-p^(-1)>_bを計算
<-q^(-1)>_bを計算
S2-q
S3-p
bp=B^2mod pを計算、bpを<bp>に変換
bq=B^2mod qを計算、bqを<bq>に変換
S3-q
S4-p
pinvを<pinv>に変換
qinvを<qinv>に変換
S4-q
S5-p
Cp=C mod pを計算、Cpを<Cp>に変換
Cq=C mod qを計算、Cqを<Cq>に変換
S5-q
S6-p
<Cp'>=MM(<Cp>,<bp>,p,aUb)を計算
S6-q
<Cq'>=MM(<Cq>,<bq>,q,aUb)を計算
S7-p
<mp'>=MEXP(<Cp'>,dp,p,aUb)を計算
S7-q
<mq'>=MEXP(<Cq'>,dq,q,aUb)を計算
S8-p
<tp>=MM(<mp'>,<qinv>,p,aUb)を計算
S8-q
<tq>=MM(<mq'>,<pinv>,q,aUb)を計算
S9-p
<up>=MUL(<tp>,<q>,aUb)を計算
S9-q
<uq>=MUL(<tq>,<p>,aUb)を計算
S10
<m'>=ADD(<up>,<uq>,aUb)を計算
S11
<m'>をm'に変換
S12
m'をmにコピー
S13
m'=m'-Nを計算
S14
m'<0
No
Yes
S15
mを出力
END

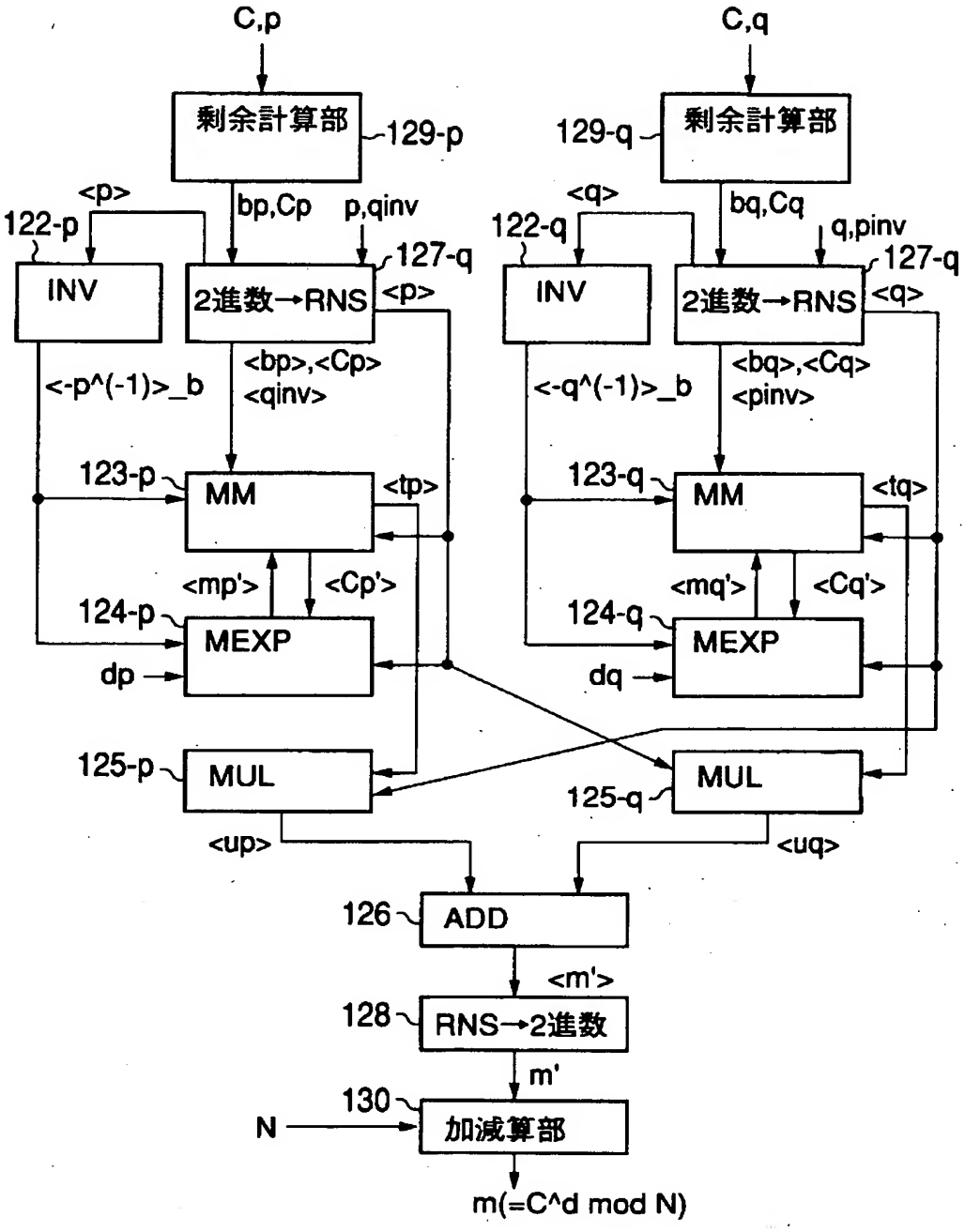
【図３】


C,p,q
剰余計算部
129
<p>,<q>
122
bp,bq,Cp,Cq
pinv,qinv
p,q
RNS逆元計算部
(INV)
2進数表現→RNS表現変換部
127
<-p^(-1)>_b
<-q^(-1)>_b
<bp>,<bq>,<Cp>,<Cq>,<pinv>,<qinv>
<p>,<q>
123
RNSモンゴメリ乗算部
(MM)
<tp>,<tq>
<mp'>,<mq'>
<Cp'>,<Cq'>
124
RNSモンゴメリべき乗部
(MEXP)
dp,dq
125
RNS乗算部(MUL)
<up>,<uq>
126
RNS加算部(ADD)
<m'>
128
RNS表現変換部→2進数表現
m'
N
加減算部
130
m(=C^d mod N)

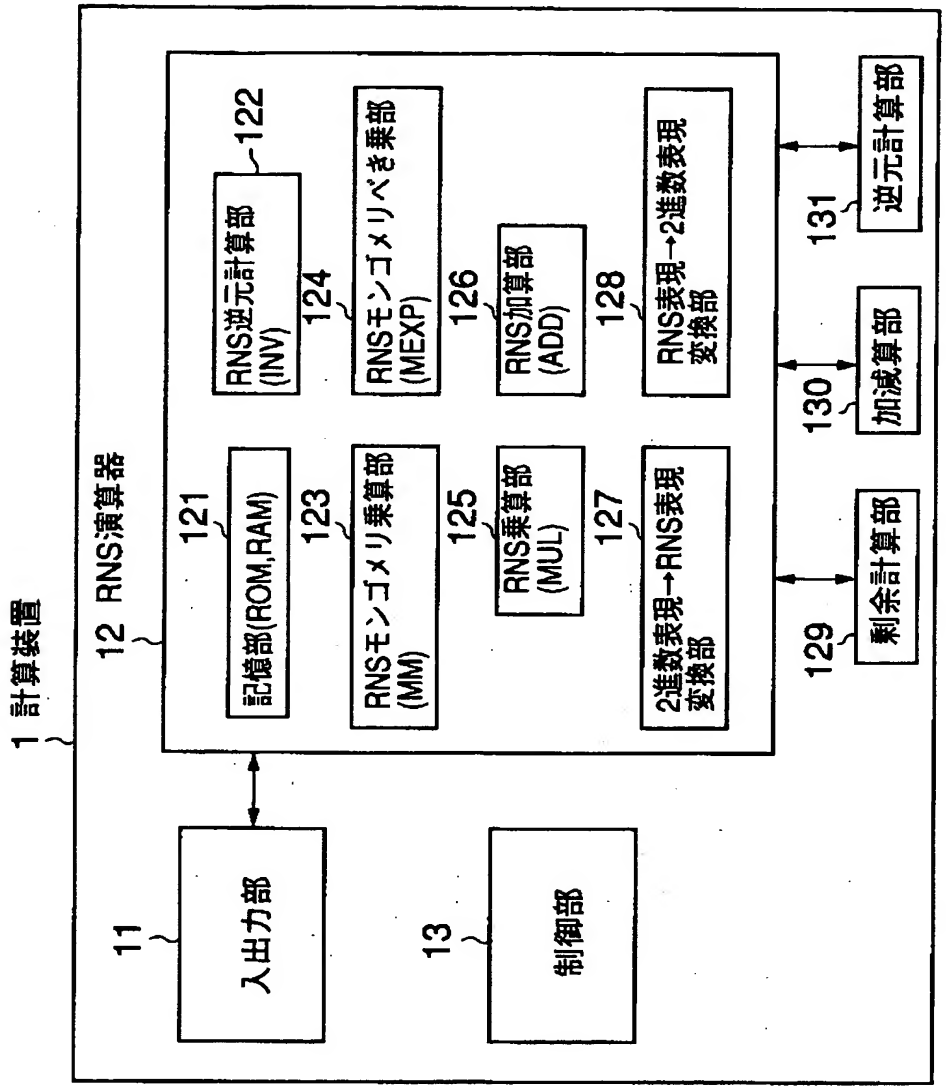
【図４】



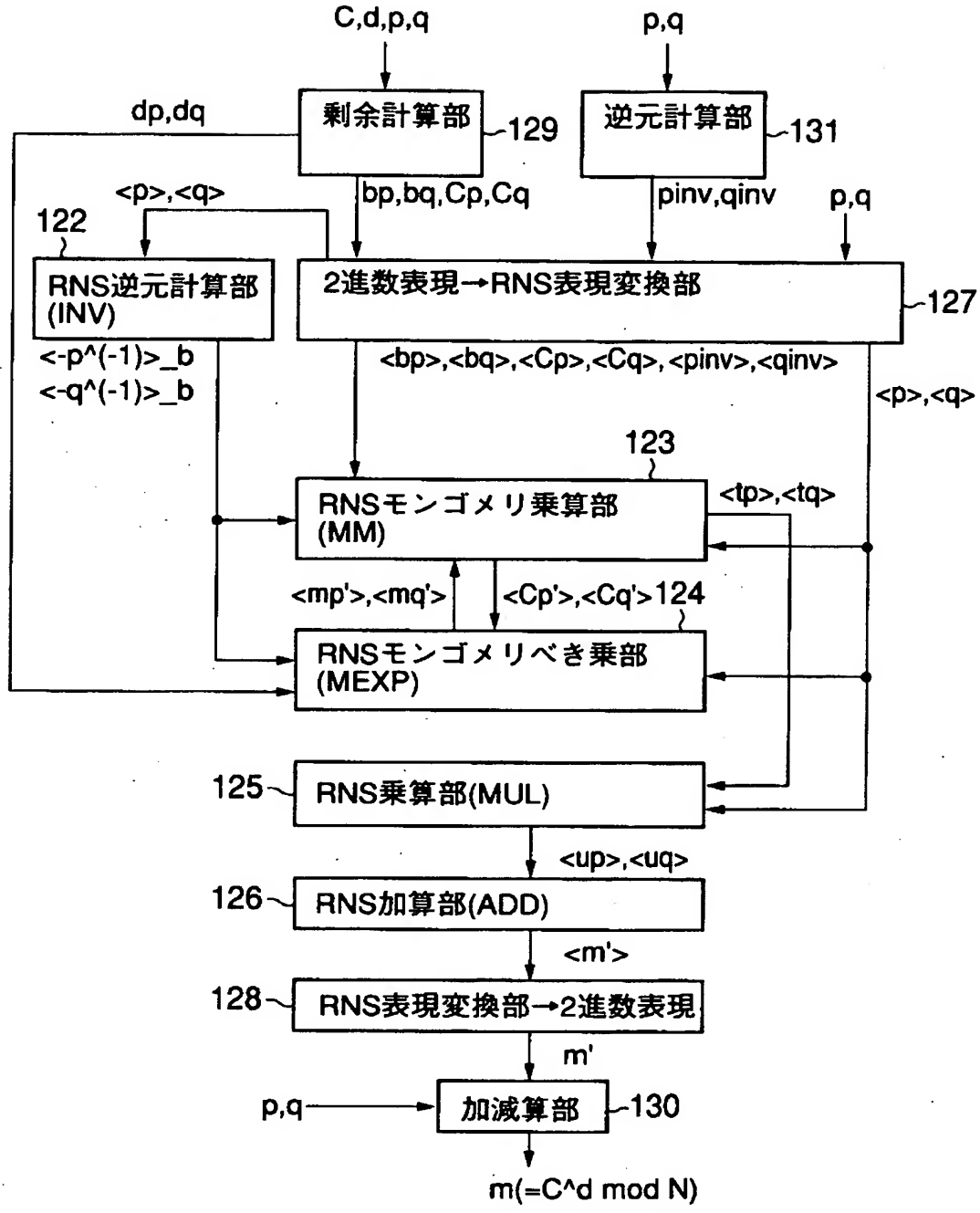
【図５】


C,p
剰余計算部
129-p
<p>
122-p
bp,Cp
p,qinv
127-q
INV
2進数→RNS
<p>
<-p^(-1)>_b
<bp>,<Cp>
<qinv>
123-p
MM
<tp>
<mp'>
<Cp'>
124-p
MEXP
dp
125-p
MUL
<up>
C,q
剰余計算部
129-q
<q>
122-q
bq,Cq
q,pinv
127-q
INV
2進数→RNS
<q>
<-q^(-1)>_b
<bq>,<Cq>
<pinv>
123-q
MM
<tq>
<mq'>
<Cq'>
124-q
MEXP
dq
125-q
MUL
<uq>
126
ADD
<m'>
128
RNS→2進数
m'
130
N
加減算部
m(=C^d mod N)

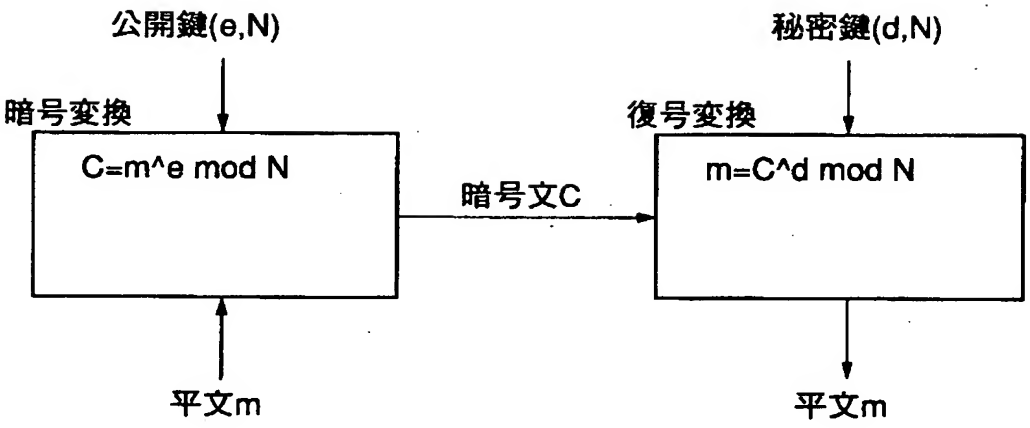
【図６】


1 計算装置
12 RNS演算器
121
記憶部(ROM,RAM)
122
RNS逆元計算部
(INV)
123
RNSモンゴメリ乗算部
(MM)
124
RNSモンゴメリべき乗部
(MEXP)
125
RNS乗算部
(MUL)
126
RNS加算部
(ADD)
127
2進数表現→RNS表現
変換部
128
RNS表現→2進数表現
変換部
129
剰余計算部
130
加減算部
131
逆元計算部
11
入出力部
13
制御部

【図７】


C,d,p,q
p,q
dp,dq
剰余計算部
129
逆元計算部
131
<p>,<q>
bp,bq,Cp,Cq
pinv,qinv
p,q
122
RNS逆元計算部
(INV)
2進数表現→RNS表現変換部
127
<-p^(-1)>_b
<-q^(-1)>_b
<bp>,<bq>,<Cp>,<Cq>,<pinv>,<qinv>
<p>,<q>
123
RNSモンゴメリ乗算部
(MM)
<tp>,<tq>
<mp'>,<mq'>
<Cp'>,<Cq'>124
RNSモンゴメリべき乗部
(MEXP)
125
RNS乗算部(MUL)
<up>,<uq>
126
RNS加算部(ADD)
<m'>
128
RNS表現変換部→2進数表現
m'
p,q
加減算部
130
m(=C^d mod N)

【図８】



【書類名】　　要約書

【要約】

【課題】　より効率良く実行可能にしたべき乗剰余計算装置を提供すること。

【解決手段】　暗号文Ｃに対し、第１及び第２の基底による剰余系表現を利用して（両基底に含まれる全整数は互いに素、第１の基底の全整数の積Ａ，第２の基底の全整数の積Ｂ＞ｐ，ｑ、Ａ＊Ｂ＞Ｃとする）、平文ｍ＝Ｃ＾ｄ　ｍｏｄ　ｐ＊ｑを求める。まず、ｐについてＣ　ｍｏｄ　ｐの剰余系表現とｄ　ｍｏｄ　（ｐ－１）に基づき（Ｃｐ＾ｄｐ）＊Ｂ　ｍｏｄ　ｐ又はこれにｐを加算した値の剰余系表現を求め、ｑについても同様にする（１２３、１２４）。得られた両剰余系表現に基づき法ｐ＊ｑの元でＣ＾ｄと合同であるｍ’の剰余系表現を求める（１２３～１２６）。ｍ’の剰余系表現を２進数表現に変換する（１２８）。得られたｐ＊ｑ未満のｍ’の値又はｐ＊ｑ以上のｍ’から所定回数ｐ＊ｑを減じることによって得たｐ＊ｑ未満の値を平文ｍとして出力する（１３０）。

【選択図】　　図３

出　願　人　履　歴　情　報

識別番号　　〔０００００３０７８〕

１．変更年月日　　１９９０年　８月２２日
［変更理由］　新規登録
住　所　神奈川県川崎市幸区堀川町７２番地
氏　名　株式会社東芝

２．変更年月日　　２００１年　７月　２日
［変更理由］　住所変更
住　所　東京都港区芝浦一丁目１番１号
氏　名　株式会社東芝